Using VisualDSolve to analyze nonlinear differential equations

P. Aaron Lott

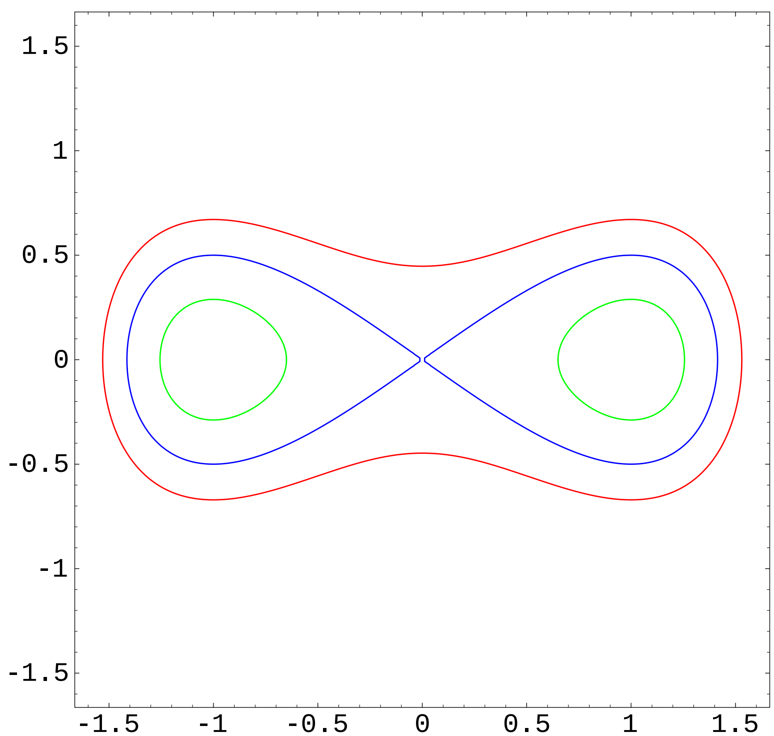
Department of Mathematics
University of Southern Mississippi

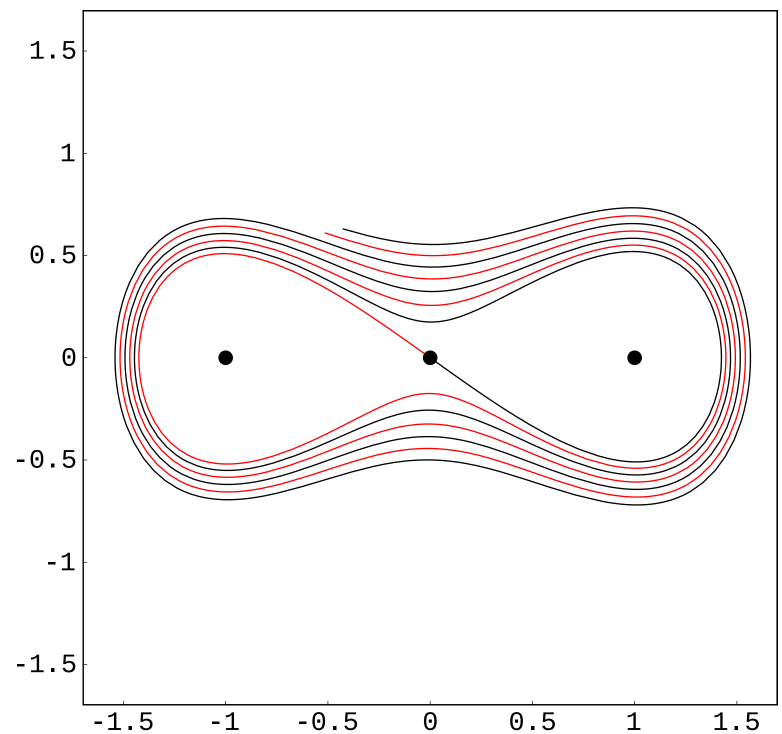
Abstract

Many nonlinear differential equations are used to model physical phenomenon, and thus there is considerable interest in knowing how to predict the behavior of solutions to a nonlinear equation. Such an understanding is often obtained by doing phase plane analysis. Using the computer algebra system *Mathematica* and an add-on package *VisualDSolve*, we investigate various methods to predict the behavior of solutions to undamped autonomous and damped autonomous nonlinear second order differential equations, through the use of contour plots, numerical solutions, and more sophisticated graphics programs.

VisualDSolve

VisualDSolve is a Mathematica add-on package designed to to analyze nonlinear differential equations visually. This package was written by Professors Stan Wagon and Dan Schwalbe of Macalester College in St. Paul Minnesota.





Nonlinear differential equations arise in models for:

Motion - Pendulum

Population - Predator Prey

Magnetic behavior - Duffing's equation

Many other mechanical and natural phenomenon

In this presentation we will demonstrate tools is to show tools that aid in the analysis of the global behavior of the following types of systems:

Undamped autonomous homogeneous

$$\ddot{x} + f(x) = 0 \tag{1}$$

Damped autonomous homogeneous

$$\ddot{x} + f(\dot{x}, x) = 0 \tag{2}$$

Non - Damped case:

The classical method of visually analyzing such systems is in the phase plane $(x, \dot{x})$.
For an example of the analysis of eq: 1 we consider the equation

$$\ddot{x} - \frac{x}{2} + \frac{x^3}{2} = 0 \tag{3}$$

Two ways we derive the phase plane of this system is via

Energy approach

Phase Plot approach

Energy approach

$$\ddot{x} - \tfrac{x}{2} + \tfrac{x^3}{2} = 0$$

Multiply through by $\dot{x}$ and integrate

$$\frac{\dot{x}^2}{2} - \frac{x^2}{4} + \frac{x^4}{8} = C \tag{4}$$

Treating $\frac{\dot{x}^2}{2}$ as kinetic energy, $-\frac{x^2}{4} + \frac{x^4}{8}$ as potential energy, and setting $\dot{x} = y$, our trajectories in the phase plane can be thought of as contours in the Energy Surface.

$$E(x, y) = \frac{y^2}{2} - \frac{x^2}{4} + \frac{x^4}{8} \tag{5}$$

Critical values of the system turn out to be the same as the critical values of $E(x, y)$. $(\pm 1, 0), (0, 0)$

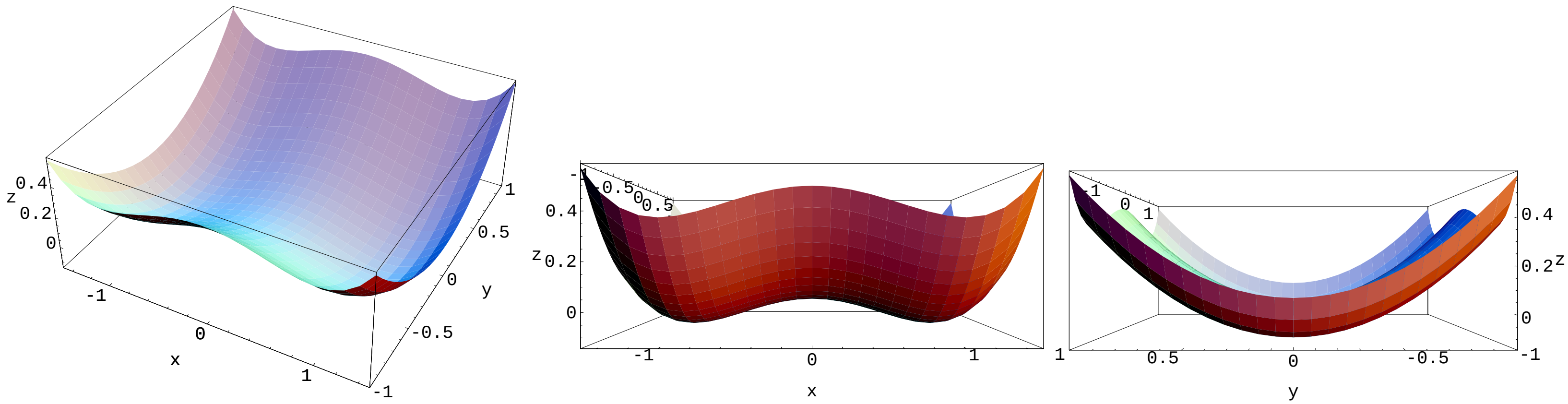


Figure 1: Various views of the Energy surface

$E(\pm 1, 0) = \frac{-1}{8}$, and $E(0, 0) = 0$. Plotting the contour curves at these critical values gives us our phase portrait.

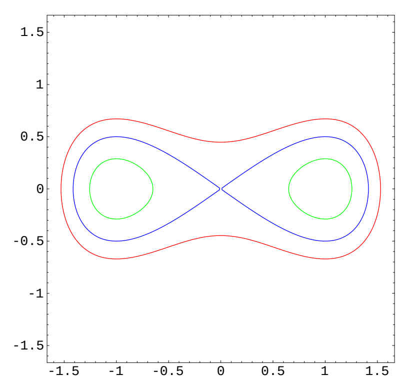


Figure 2: Energy levels at $\frac{-1}{8}, \frac{-1}{12}, 0, 0.1$.

Here we can see our separatrix and classify the behavior of any trajectory noting the position of the initial conditions.

Vector fields

Vector fields are often used to visualize the behavior inside of the phase plane. One can see how the "fish" from VisualDSolve's FlowField command produces a better perception of the trajectories in the following example.

```
PhasePlot[{x'[t] == y[t], y'[t] == .5*x[t] - .015*x'[t]-.5*x[t]^3},
   {x[t], y[t]}, {t, 0,22}, {x, -1.7, 1.7}, {y,-.75,.75},
   FlowField -> True, NumberFish -> 20, Segments -> 10,
   MaxSteps ->Infinity, WorkingPrecision -> 28];

PhasePlot[{x'[t] == y[t], y'[t] == .5*x[t] - .5*x[t]^3}, {x[t], y[t]},
   {t, 0, 22}, {x, -1.7, 1.7}, {y, -.75, .75}, VectorField -> True,
   MaxSteps -> Infinity, WorkingPrecision -> 28]
```

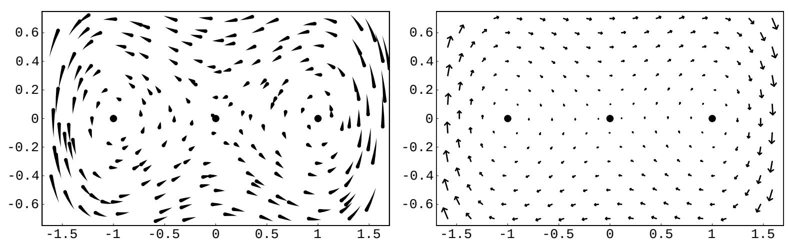


Figure 3: Phase Portrait with fish vs. vector field.

PhasePlot approach

The same type of analysis can be performed using VisualDSolve with the PhasePlot command. Given a system, initial conditions, and a time interval, PhasePlot will produce the phase portrait of the system.

```
PhasePlot[{x'[t] == y[t], y'[t] == .5*x[t]  -.5*x[t]^3},
    {x[t], y[t]}, {t,0,22}, {x, -1.7, 1.7},{y, - .75,  .75},
    InitialValues -> {{.5,.2}, {- .5,  .2}, {-.5,-0.33}, {.5,0.33},
    {-.42, .53}}, ParametricPlotFunction -> FlowParametricPlot,
    Rainbow ->True, NumberFish -> 120, Segments ->5,MaxSteps -> Infinity,
    WorkingPrecision -> 28];
```

Here we show PhasePlot's FlowParametricPlot option. This option shows part of the vector field plots small "fish" to show the varying speed of the trajectory over the phase plane.

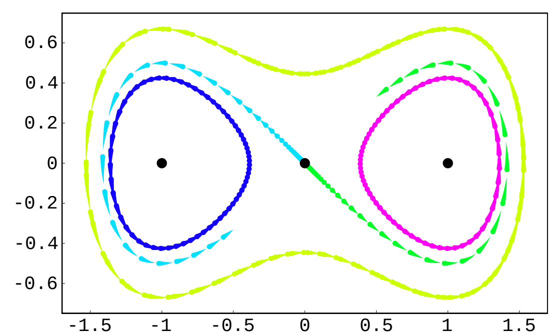


Figure 4: Trajectories using fish.

Animation 1

Damped case:

Again, we would like to have a method to generate the phase plane of the system. For this case, we consider the equation

$$\ddot{x} + \frac{15}{1000}\dot{x} - \frac{x}{2} + \frac{x^3}{2} = 0 \tag{6}$$

The energy approach used in the non-damped case is no longer applicable, since we can not integrate the $\dot{x}^2$ term. For the analysis of this equation we once again turn to VisualDSolve's PhasePlot method.

PhasePlot

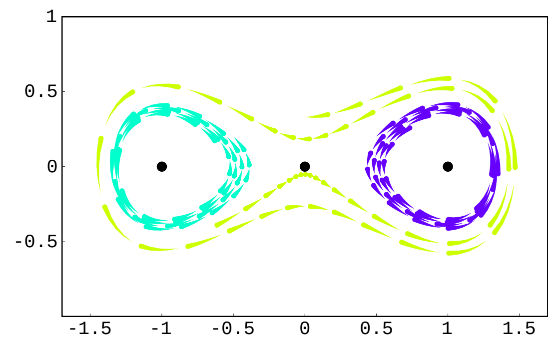


Figure 5: Trajectories using PhasePlot

We now analyze the behavior of the system by the phase plane. Notice that the centers we saw in figure 4 have now become attracting spiral points. Note separatrix is difficult to see.

Animation2

Basins of Attraction As in the non-damped case, we would like to be able to derive a separatrix so

that we can separate regions called basins of attraction. In the VisualDSolve book, the authors present a way to approximate the separatrix. One picks two points near the origin and traces their trajectories over negative time intervals.

```
PhasePlot[{x'[t] == y[t], y'[t] == .5*x[t] - .015*x'[t] - .5*x[t]^3},
{x[t], y[t]}, {t, -0.1, -60}, {x,-1.7,1.7}, {y,-.75,.75},
InitialValues -> {{0, - .00001}, {0,  .00015}},
ShowEquilibria -> True, PlotStyle -> {{White},{Black}},
WindowShade -> GrayLevel[0.5], MaxSteps ->Infinity]
```

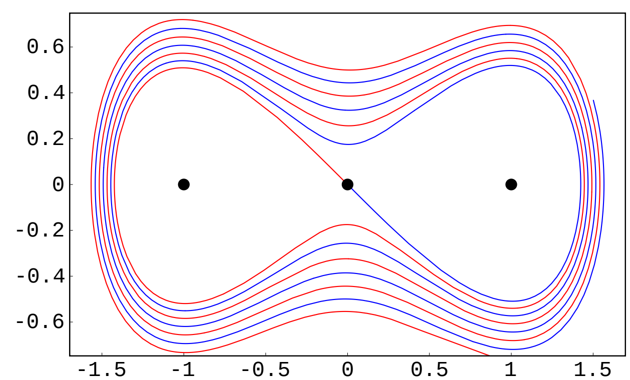


Figure 6: "Separatrix" of eq.(6) using PhasePlot.

Separatrix Approximation

PhasePlot creates an interpolating function using Mathematica's NDSolve. So, if we pick initial points p_1 and p_2 from the interpolating function generated in the previous code and run PhasePlot over a positive time interval, we should still have an approximation to the separatrix

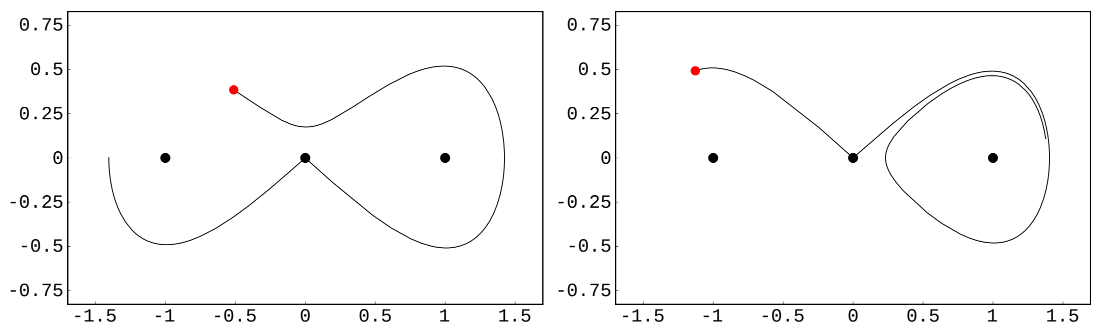


Figure 7: Trajectories obtained by running PhasePlot over the positive time interval $(0, 80)$

However, as we can see in the plots above, our points from the interpolating function do not lie on the separatrix since their trajectories fall into the basins.

Better Approximation

By the geometry of the phase plane, we can see that if we pick two points p_1, p_2 that are close to each other, with trajectories that lead to opposite basins of attraction, then there is a point p^* between them that lies on the separatrix. One may now use the bisection method on the line segment joining (p_1, p_2) and search for the point p^*.

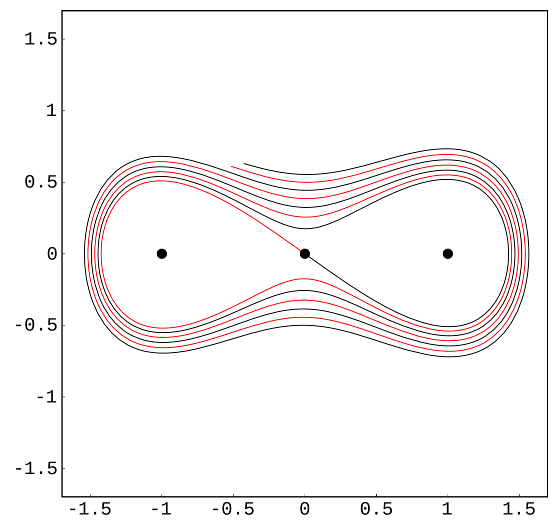


Figure 8:

Trajectories obtained by running PhasePlot over the positive time interval (0,80) with initial conditions found using a trial and error root finding algorithm.

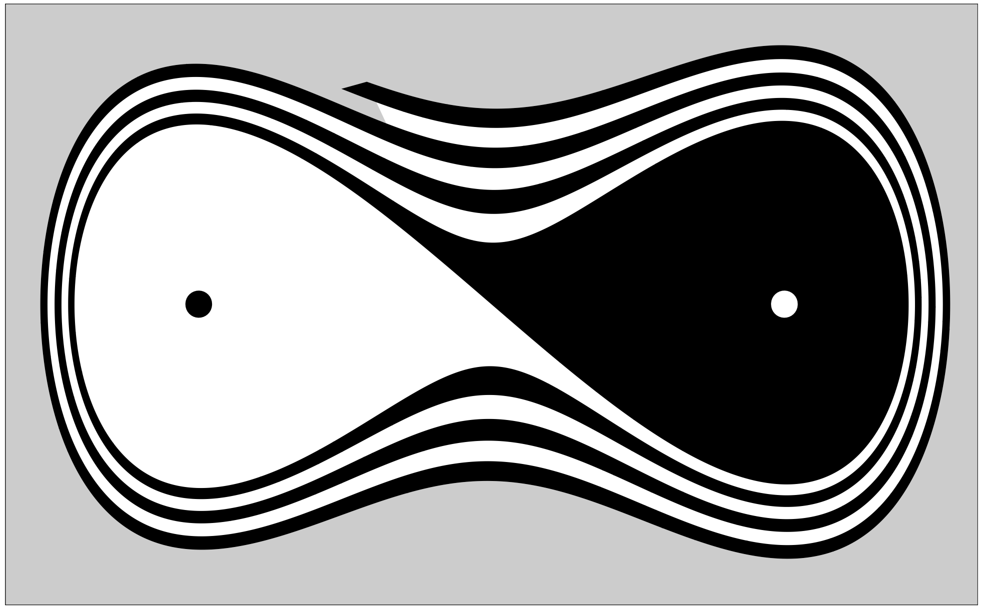


Figure 9: Two distinct basins of attraction generated by polygon programming in the VisualDSolve text

Conclusion

There is a reform under way in the teaching of elementary differential equations which down plays the solving of specific types of equations and emphasizes qualitative aspect and nonlinear equations. Computer oriented projects and visualization are the heart of the reform. We have been investigating how to display the nature of critical values, trajectories, phase portraits, and other features of a nonlinear equation so that one can effortlessly analyze the global behavior of dynamical systems.

We can now:

Use the energy approach to obtain a global view of non damped systems

Generate phase portraits of dynamical systems

Create animations of the phase portraits

Analyze phase plane of nonlinear differential equations